Boundary conditions control in ORCA2

Eugene Kazantsev

Boundary conditions control in ORCA2

Eugene Kazantsev

INRIA, Moise

Journée thématique
"Que peuvent attendre les modélisateurs de l'assimilation de données ?"
Paris, le 12 février 2013

ORCA-2 configuration of NEMO

- $182 \times 149 \times 31$ nodes in curvilinear (x, y) coordinates;
- z levels with partial steps at the bottom;
- leap-frog scheme with Asselin filter;
- implicit surface pressure gradient with External Gravity Waves filter;
- implicit vertical diffusion with TKE Turbulent Closure Scheme;
- Solar Radiation + Geothermal Heating + BBL + Surface evaporation/precipitation;
- surface wind stress.

Space discretization on the C-grid

$$
\begin{aligned}
\frac{\partial u}{\partial t} &= \Big(S_x S_y v\Big) S_y(\omega + f) - D_x \frac{S_x u^2 + S_y v^2}{2} - S_z\Big(S_x w D_z u\Big) + D_x A_u^h \xi + D_y A_u^h \omega + \\
&+ g \int_0^z D_x S_z \rho(x,y,\zeta) d\zeta + D_{zz}(A_u^z u) + g D_x(\eta + T_c \phi) \\
\frac{\partial T}{\partial t} &= -D_x(u S_x T) - D_y(v S_y T) - D_z(w S_z T) + A_T^h \Big(D_x D_x T + D_y D_y T\Big) + \\
&+ D_{zz}(A_T^z T) + \text{Solar Radiation} + \text{Geothermal Heating} + \text{BBL} \\
\xi &= D_x u + D_y v, \quad \omega = D_y u - D_x v, \quad w = \int_H^z \xi(x,y,\zeta) d\zeta; w(x,y,H) = 0
\end{aligned}
$$

Interpolations and Derivatives

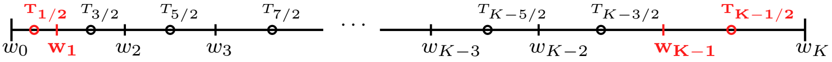
$$
\begin{aligned}
(Sw)_{k+1/2} &= \frac{w_{k+1} + w_k}{2} \; k = 1, \ldots, K-1 \\
(DT)_k &= \frac{T_{k+1/2} - T_{k-1/2}}{h} \; k = 1, \ldots, K-1
\end{aligned}
$$

Space discretization on the C-grid

$$
\begin{aligned}
\frac{\partial u}{\partial t} &= \Big(S_x S_y v\Big) S_y(\omega + f) - D_x \frac{S_x u^2 + S_y v^2}{2} - S_z \Big(S_x w D_z u\Big) + D_x A_u^h \xi + D_y A_u^h \omega + \\
&+ g \int_0^z D_x S_z \rho(x,y,\zeta) d\zeta + D_{zz}(A_u^z u) + g D_x(\eta + T_c \phi) \\
\frac{\partial T}{\partial t} &= -D_x(u S_x T) - D_y(v S_y T) - D_z(w S_z T) + A_T^h \Big(D_x D_x T + D_y D_y T\Big) + \\
&+ D_{zz}(A_T^z T) + \text{Solar Radiation} + \text{Geothermal Heating} + \text{BBL} \\
\xi &= D_x u + D_y v, \quad \omega = D_y u - D_x v, \quad w = \int_H^z \xi(x,y,\zeta) d\zeta; w(x,y,H) = 0
\end{aligned}
$$

Interpolations and Derivatives Modified Near the boundary

$$
\begin{aligned}
(Sw)_{k+1/2} &= \frac{w_{k+1} + w_k}{2}, \ k = 1, \dots, K-2, \ (Sw)_{1/2} = \alpha_0^S + \alpha_1^S w_0 + \alpha_2^S w_1 \\
(DT)_k &= \frac{T_{k+1/2} - T_{k-1/2}}{h}, \ i = 2, \dots, K-2, \ (DT)_1 = \alpha_0^D + \frac{\alpha_1^D T_{1/2} + \alpha_2^D T_{3/2}}{h}
\end{aligned}
$$

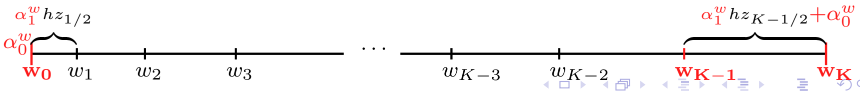


Space discretization on the C-grid

$$
\begin{aligned}
\frac{\partial u}{\partial t} &= \Big(S_x S_y v\Big) S_y(\omega + f) - D_x \frac{S_x u^2 + S_y v^2}{2} - S_z \Big(S_x w D_z u\Big) + D_x A_u^h \xi + D_y A_u^h \omega + \\
&+ \quad g \int_0^z D_x S_z \rho(x,y,\zeta) d\zeta + D_{zz}(A_u^z u) + g D_x(\eta + T_c \phi) \\
\frac{\partial T}{\partial t} &= -D_x(u S_x T) - D_y(v S_y T) - D_z(w S_z T) + A_T^h \Big(D_x D_x T + D_y D_y T\Big) + \\
&+ \quad D_{zz}(A_T^z T) + \text{Solar Radiation} + \text{Geothermal Heating} + \text{BBL} \\
\xi &= D_x u + D_y v, \quad \omega = D_y u - D_x v, \quad w = \int_H^z \xi(x,y,\zeta) d\zeta; w(x,y,H) = \alpha_0(x,y)
\end{aligned}
$$

Vertical velocity

$$
\begin{aligned}
w_{i,j,K-1} &= \alpha_0^{w^b} - \alpha_1^{w^b} hz_{i,j,K-1/2}\xi_{i,j,K-1/2} \\
w_{i,j,k-1} &= w_{i,j,k} - hz_{i,j,k-1/2}\xi_{i,j,k-1/2} \quad \forall k : 2 \le k \le K-1 \\
w_{i,j,0} &= w_{i,j,1} + \alpha_0^{w^s} - \alpha_1^{w^s} hz_{i,j,1/2}\xi_{i,j,1/2}
\end{aligned}
$$



Vertical diffusion

$$\frac{\partial}{\partial z} A_u^z \frac{\partial u}{\partial z} \quad \text{is replaced by}$$

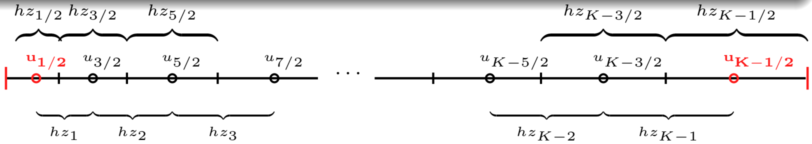
$$\left(D_{zz}u\right)_{i,j,1/2} = \frac{(A_u^z)_1}{hz_1 hz_{1/2}}(\alpha_2^{DzzU^s} u_{3/2} - \alpha_1^{DzzU^s} u_{1/2})$$

$$\left(D_{zz}u\right)_{i,j,k-1/2} = \frac{1}{hz_{k-1/2}}\left(\frac{(A_u^z)_k}{hz_k}(u_{k+1/2} - u_{k-1/2}) - \frac{(A_u^z)_{k-1}}{hz_{k-1}}(u_{k-1/2} - u_{k-3/2})\right) \quad \forall k : 2$$

$$\left(D_{zz}u\right)_{i,j,K-1/2} = \frac{1}{hz_{K-1/2}}\left[\alpha_2^{DzzU^b} \frac{(A_u^z)_{K-1}}{hz_{K-1}} u_{K-1/2} - \alpha_1^{DzzU^b}\left(\frac{(A_u^z)_K}{hz_K} + \frac{(A_u^z)_{K-1}}{hz_{K-1}}\right) u_{K-3}\right.$$

$$\left.\frac{\partial u}{\partial z}\right|_{w0} = \alpha_0^{DzzU^s} + \frac{\tau_x}{hz_1 \rho_0}, \quad \left.\frac{\partial v}{\partial z}\right|_{w0} = \alpha_0^{DzzU^s} + \frac{\tau_y}{hz_1 \rho_0}, \quad \left.\frac{\partial T}{\partial z}\right|_{w0} = \left.\frac{\partial S}{\partial z}\right|_{w0} = \alpha_0^{DzzT^s}$$

$$u|_{bottom} = v|_{bottom} = \alpha_0^{DzzU^b} \quad T|_{bottom} = S|_{bottom} = \alpha_0^{DzzT^b} \tag{1}$$



The models solution depend on initial and boundary conditions :

$$\frac{\partial T}{\partial t} = -D_x(uS_xT) - D_y(vS_yT) - D_z^{(\alpha)}(wS_z^{(\alpha)}T) + A_T^h\left(D_{xx}T + D_{yy}T\right) + Dzz^{(\alpha)}(A_T^z)T$$

- The model $x(t) = \mathcal{M}_{0,t}(x_0, \alpha)$

We calculate the derivatives and their adjoints with respect to

$$x_0, \quad \alpha$$

by TAPENADE 3.6 (Tropics team, INRIA) that allows us

- to avoid a HUGE development/coding (a double of the classical one, at least)
- to obtain immediately the derivative with respect to any parameter we want.

TAPENADE 3.6 (Tropics team, INRIA) with the Memory Usage Optimization:

search for push/pop

```
CALL PUSHREAL8ARRAY(sold, nx*ny*nz)
CALL PUSHREAL8ARRAY(told, nx*ny*nz)
CALL PUSHREAL8ARRAY(vold, nx*ny*nz)
CALL PUSHREAL8ARRAY(uold, nx*ny*nz)
CALL PUSHREAL8ARRAY(ssh, nx*ny)
CALL PUSHREAL8ARRAY(s, nx*ny*nz)
CALL PUSHREAL8ARRAY(t, nx*ny*nz)
CALL PUSHREAL8ARRAY(v, nx*ny*nz)
CALL PUSHREAL8ARRAY(u, nx*ny*nz)
```

replace by

```
call push_uvts(u,v,t,s,ssh)
```

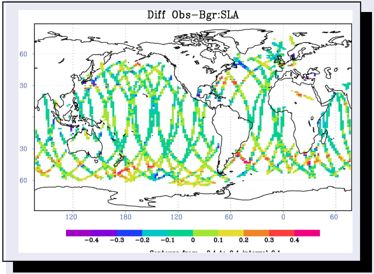
Procedure push/pop_uvts(u,v,t,s,ssh):

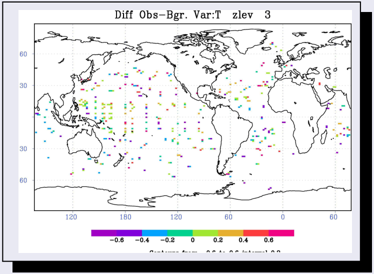
- does not push $n-1$ step and pops appropriate values (divides the required memory by 2)
- does not push u, v, t, s in lower level routines
- does not push values on continents (divides by 2)
- pushes values in Real*4 format (divides by 2)
- eventually pushes only odd timesteps and interpolate when poping (divides by 2)

Total reduction of required memory is up to 25 times.
10 hours window $\Longrightarrow$ 10 days window.

ECMWF data issued from Jason-1 and Envisat altimetric missions and ENACT/ENSEMBLES data banque.

January, 1, 2006.





Difference between observations and background during the 1st of January.

The model: $x_N = \mathcal{M}_{0,N}(x_0, \alpha)$ with $x = (u, v, T, S, ssh)^T$

Cost function J

$$
\begin{aligned}
J &= \|x_0 - x_{bgr}\|^2_{B^{-1}} + \|\alpha - \alpha_{bgr}\|^2_{B^{-1}} + \\
&+ \sum_{n=0}^{N} t_n \|\mathcal{H}\mathcal{M}_{0,n}(x_0, \alpha) - y_n\|^2_{R^{-1}}
\end{aligned}
$$

Matrices: $B^{-1} = diag(10^{-4})$,
$R^{-1} = diag(1/\sigma_u, 1/\sigma_v, 1/\sigma_T, 1/\sigma_S, 1/\sigma_{ssh})$ where $\sigma_u^2 = \frac{1}{N_{obs}} \sum (u_{obs} - u_{bgr})^2$
Minimization is performed by M1QN3 (JC Gilbert, C.Lemarechal)

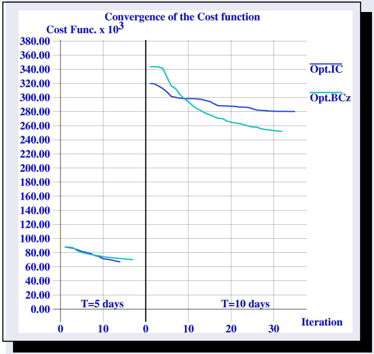
Data Assimilation – Forecast

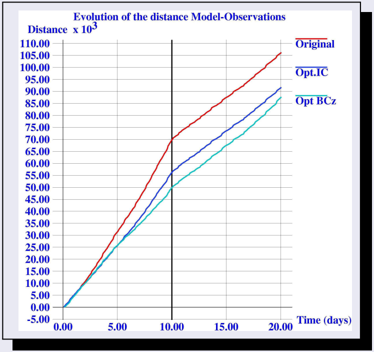
Assimilation window — 10 days (Jan. 1-10, 2006),
Test time — 20 or 30 days (Jan. 1-31, 2006).

The model: $x(t) = \mathcal{M}_{0,t}(x_0, \alpha)$ with $x = (u, v, T, S, ssh)^T$

$$\text{Distance: } \xi(t) = \sum_{n=0}^{t} \| \mathcal{H}\mathcal{M}_{0,n}(x_0, \alpha) - y_n \|_{R^{-1}}$$

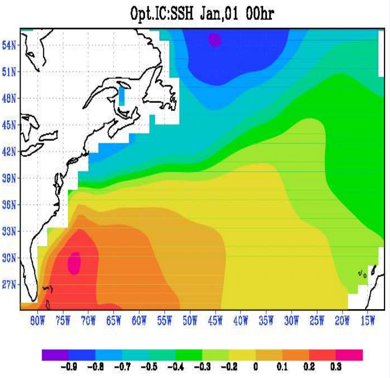
Convergence of J and evolution of ξ



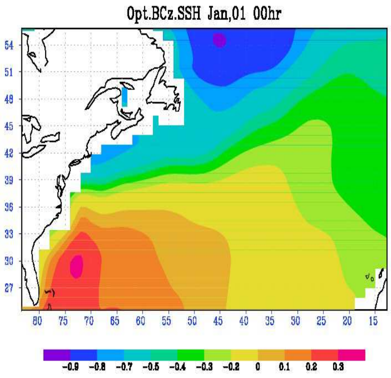


20 Cost function calls with $T = 5$ days and 40 calls with $T = 10$ days.

SSH, North Atlantic, January,1-30 2006.

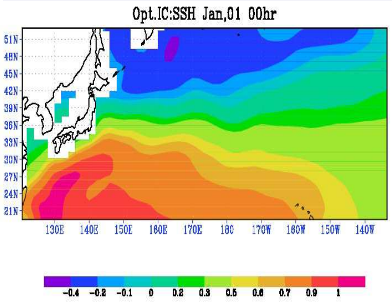


Optimal IC

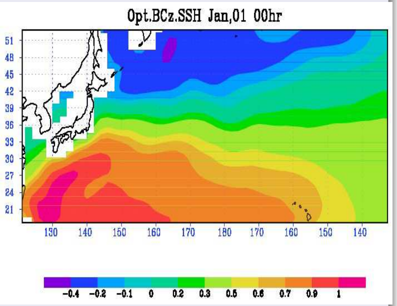


Optimal BCz

SSH, North Pacific, January,1-30 2006.



Optimal IC



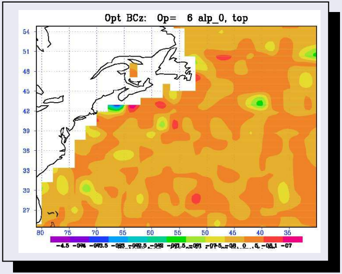
Optimal BCz

Modified formula

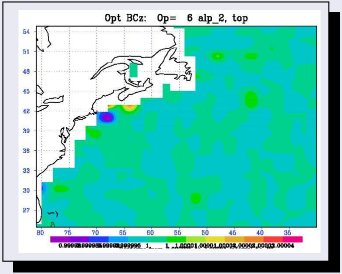
$$
\begin{aligned}
w_{i,j,K-1} &= \alpha_0^{w^b} - \alpha_2^{w^b} hz_{i,j,K-1/2}\xi_{i,j,K-1/2} \\
w_{i,j,k-1} &= w_{i,j,k} - hz_{i,j,k-1/2}\xi_{i,j,k-1/2} \quad \forall k : 1 \le k \le K-2 \\
w_{i,j,0} &= w_{i,j,1} + \alpha_0^{w^s} - \alpha_2^{w^s} hz_{i,j,1/2}\xi_{i,j,1/2}
\end{aligned}
$$

BC for the vertical velocity

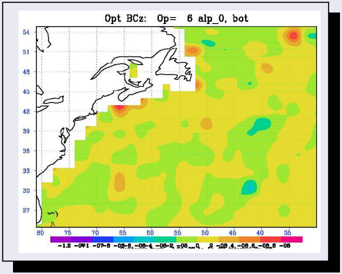
α for the vertical velocity w. North Atlantic.



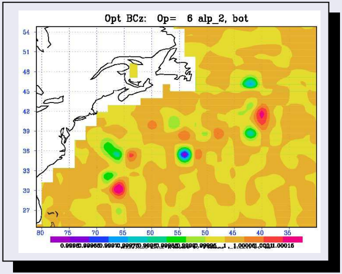
α_0 on the surface



α_2 on the surface

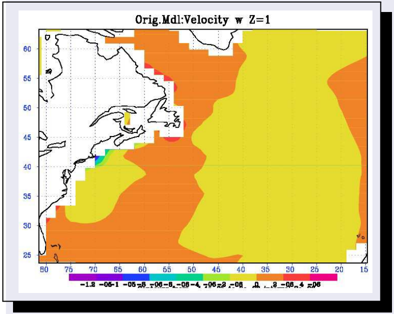


α_0 on the bottom

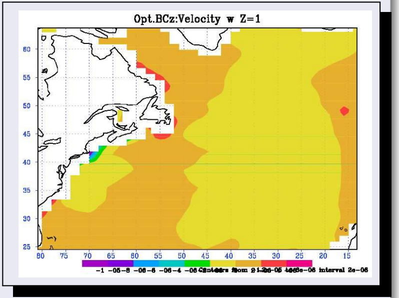


α_2 on the bottom

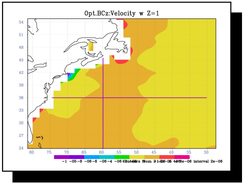
North Atlantic, January, 30, 2006, surface



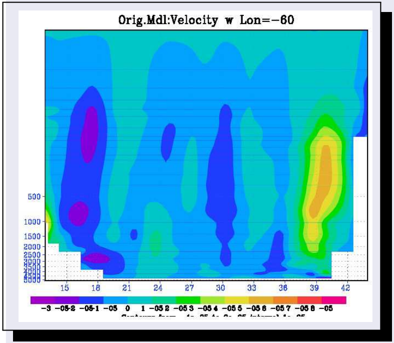
Original model



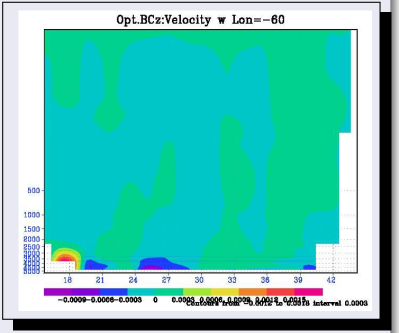
Optimal BCz



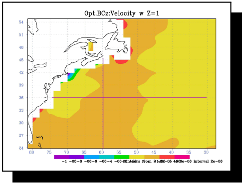
North Atlantic, January, 30, 2006, $y - z$ section



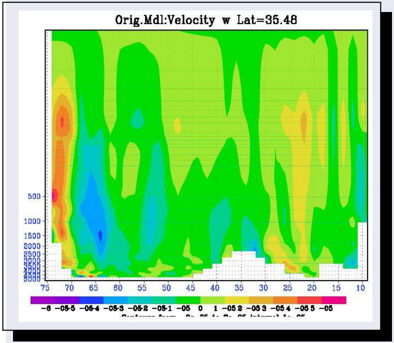
Original model



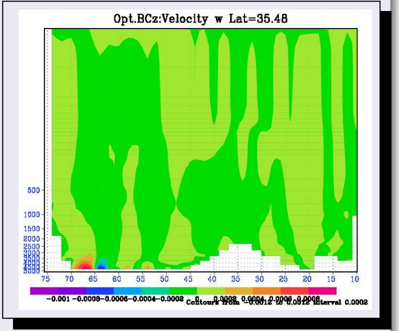
Optimal BCz



North Atlantic, January, 30, 2006, $x - z$ section

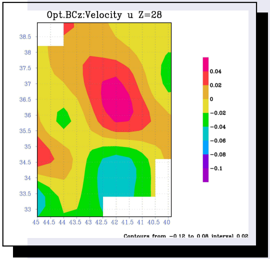


Original model

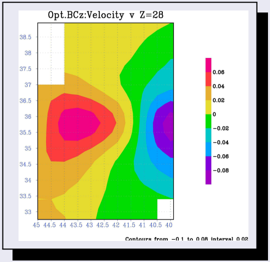


Optimal BCz

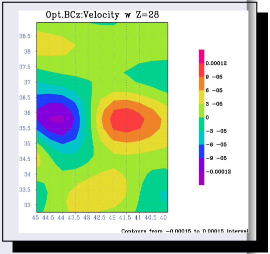
Levels $z = 28$ and $z = 29$



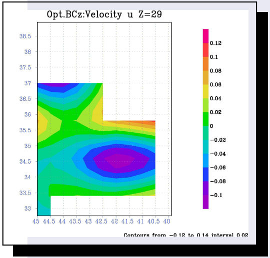
Velocity u



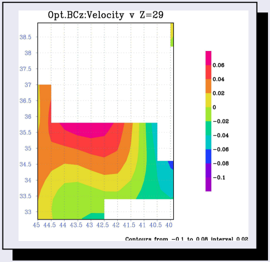
Velocity v



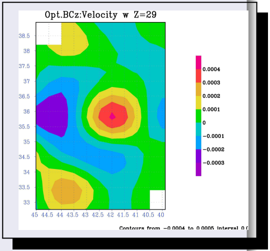
Velocity w



Velocity u



Velocity v

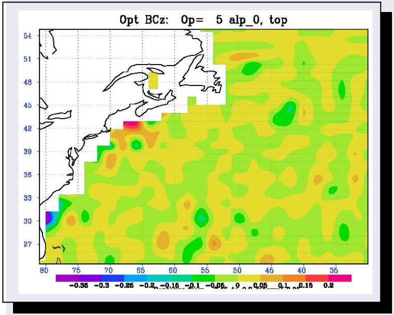


Velocity w

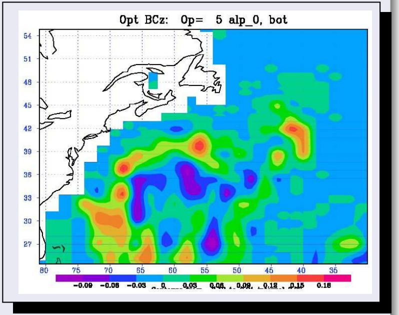
$$
\left.\frac{\partial u}{\partial z}\right|_{w_0} = \alpha_0^{DzzU^s} + \frac{\tau_x}{hz_1\rho_0}, \quad \left.\frac{\partial v}{\partial z}\right|_{w_0} = \alpha_0^{DzzU^s} + \frac{\tau_y}{hz_1\rho_0},
$$

$$
u|_{bottom} = v|_{bottom} = \alpha_0^{DzzU^b}
$$

North Atlantic

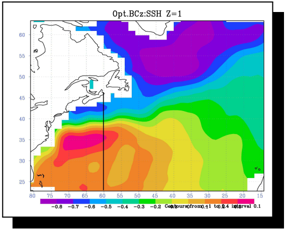


Surface

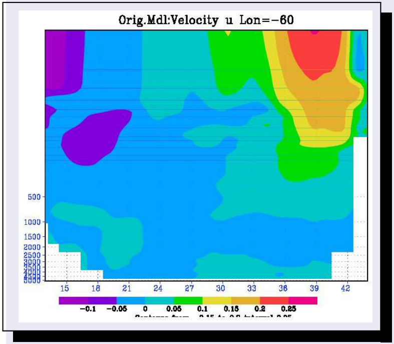


Bottom

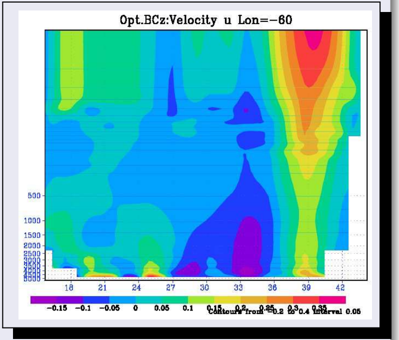
Velocity components



North Atlantic, January, 30, 2006, Velocity u, $y-z$ section



Original model



Optimal BCz

It is not an artefact.

North Atlantic

Modification of the SSH in the North Atlantic is strongly related to the boundary conditions of u and v especially on the bottom.

Only α_0 on the Bottom for u and v, only in the Vertical diffusion

$$\frac{\partial}{\partial z} A_u^z \frac{\partial u}{\partial z} \quad \text{is replaced by}$$

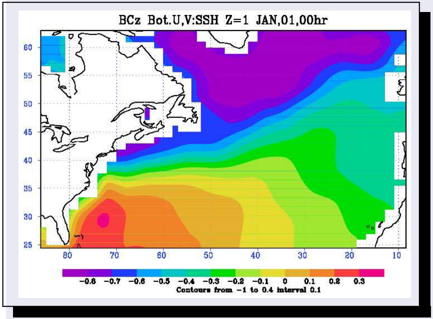
$$\begin{aligned}
\left(D_{zz}u\right)_{i,j,1/2} &= \frac{(A_u^z)_1}{hz_1 hz_{1/2}}(u_{3/2}-u_{1/2}) \\
\left(D_{zz}u\right)_{i,j,k-1/2} &= \frac{1}{hz_{k-1/2}}\left(\frac{(A_u^z)_k}{hz_k}(u_{k+1/2}-u_{k-1/2}) - \frac{(A_u^z)_{k-1}}{hz_{k-1}}(u_{k-1/2}-u_{k-3/2})\right) \quad \forall k : 2 \\
\left(D_{zz}u\right)_{i,j,K-1/2} &= \frac{1}{hz_{K-1/2}}\left[\frac{(A_u^z)_{K-1}}{hz_{K-1}}u_{K-1/2} - \left(\frac{(A_u^z)_K}{hz_K} + \frac{(A_u^z)_{K-1}}{hz_{K-1}}\right)u_{K-3/2}\right] \\
& u|_{bottom} = \alpha_0^u \quad v|_{bottom} = \alpha_0^v \qquad (2)
\end{aligned}$$

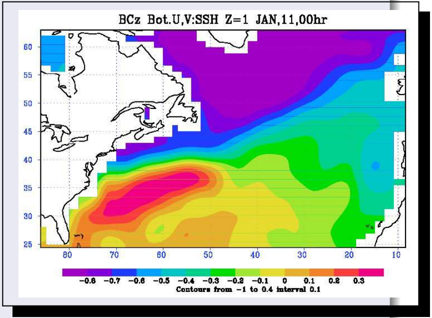
Control space dimension

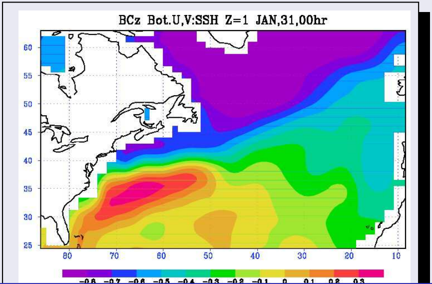
Initial conditions:	1 707 245
Full vertical boundary:	1 197 792
Only bottom:	33 272

SSH in the restrained control experiment

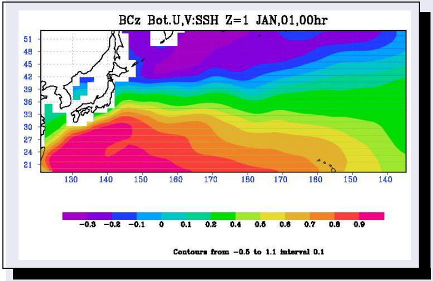
North Atlantic

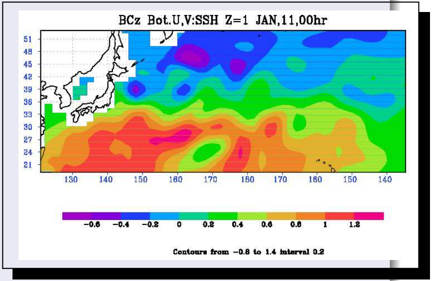


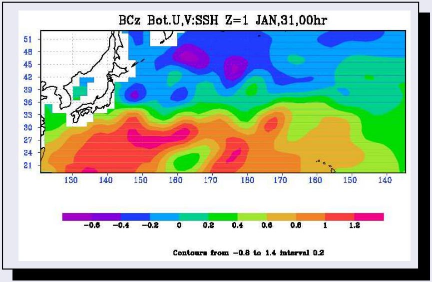




North Pacific







Extending the set of control parameters we can

- find a way to compensate model errors
- showing the most influent parameter and the most important geographical regions.

Automatic adjoint code generation helps us

- to generate TLM/AM almost immediately,
- to avoid a HUGE development/coding,
- to obtain immediately the derivative with respect to any parameter we want.

http://www-ljk.imag.fr/membres/Kazantsev/orca2/index.html